(12) **UK Patent Application** (19) **GB** (11) **2 364 461** (13) **A**

(21) Application No 0111466.9

(22) Date of Filing 10.05.2001

(30) Priority Data
(31) 09578251 (32) 24.05.2000 (33) US

(71) Applicant(s)
Hewlett-Packard Company
(Incorporated in USA - Delaware)
3000 Hanover Street, Palo Alto, California 94304,
United States of America

(72) Inventor(s)
Jonathan Yen
Daniel R Tretter
Truong-Thao Nguyen

(74) Agent and/or Address for Service
Carpmaels & Ransford
43 Bloomsbury Square, LONDON, WC1A 2RA,
United Kingdom

(51) INT CL^7
H04N 5/217 // H04N 7/68 9/88

(52) UK CL (Edition T)
H4F FHC FHHE F30J

(56) Documents Cited
GB 2164521 A   GB 2120895 A
EP 0615205 A2   EP 0478040 A2

(58) Field of Search
Online: EPODOC, WPI, PAJ

(54) Abstract Title
Correcting defective pixels in an image

(57) A source image formed from a plurality of pixels each having a respective value may have defective pixels (X). A defective pixel (X) is corrected based on curvature information computed from pixel values of pixels located near the defective pixel (X). Curvature is defined as being calculated by performing arithmetic functions on values of pixels in rows above and below a row containing the defective pixel (X) or on values of pixels in columns preceding and succeeding the column containing the defective pixel (X). A second independent claim is included which claims the correction of defective pixel values by determining a median pixel value from pixels on the same colour plane located near the defective pixel (X).
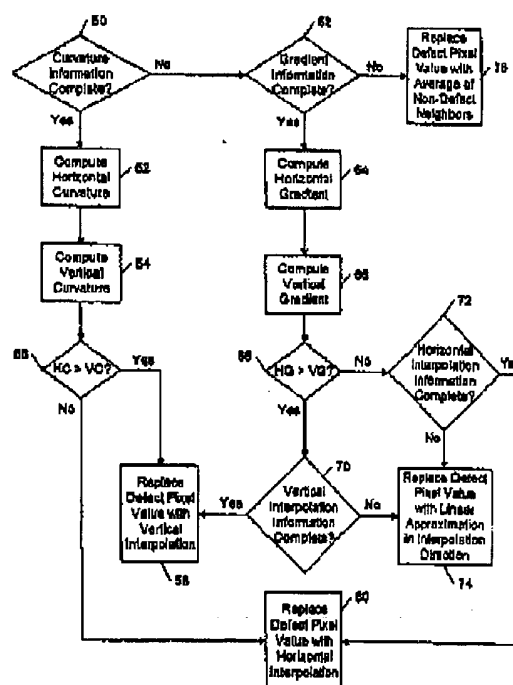
FIG. 4

FIG. 1

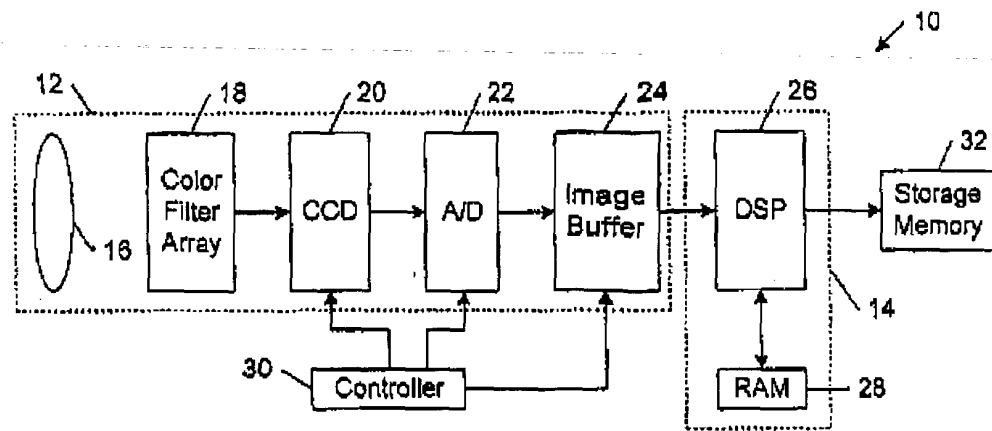FIG. 2

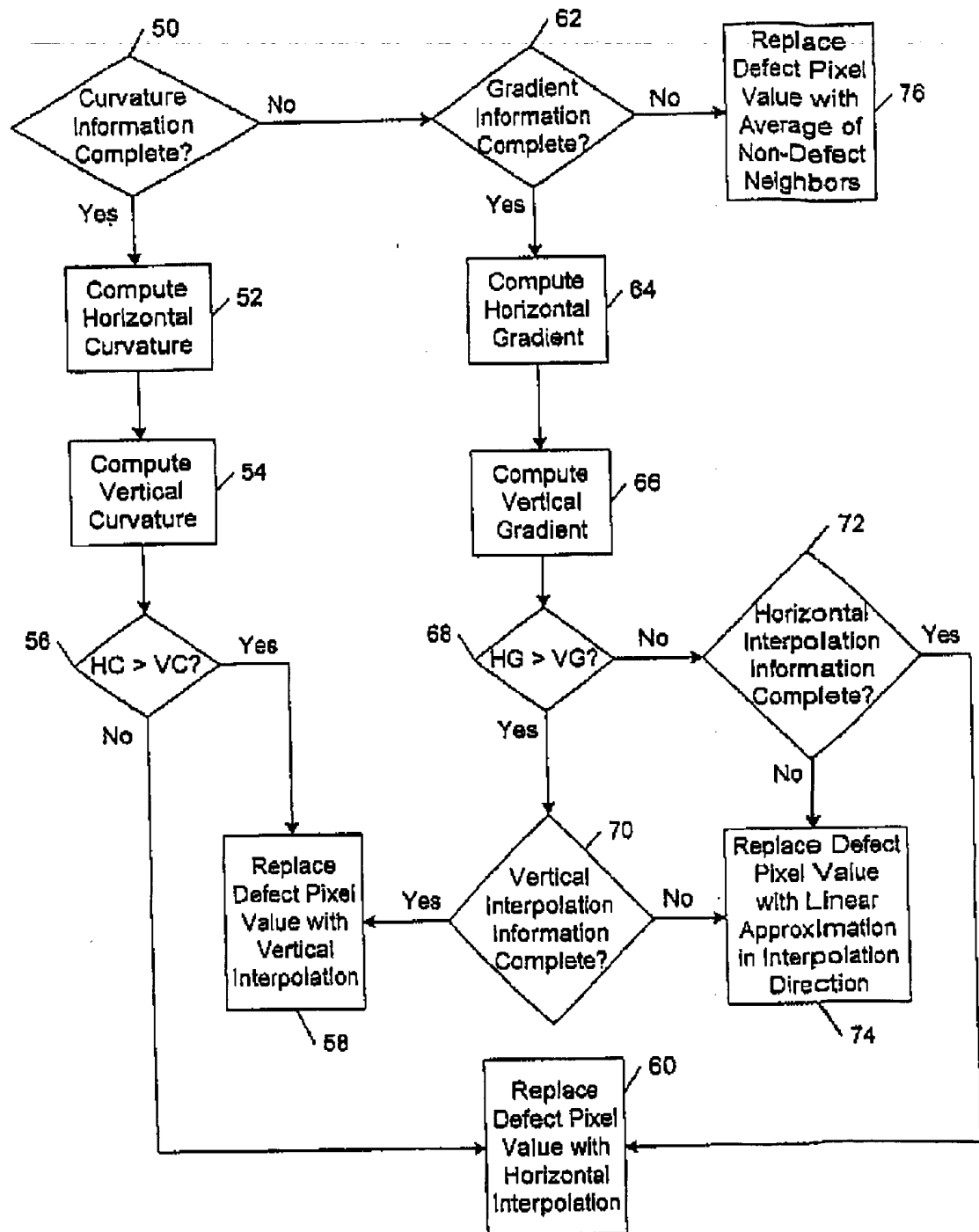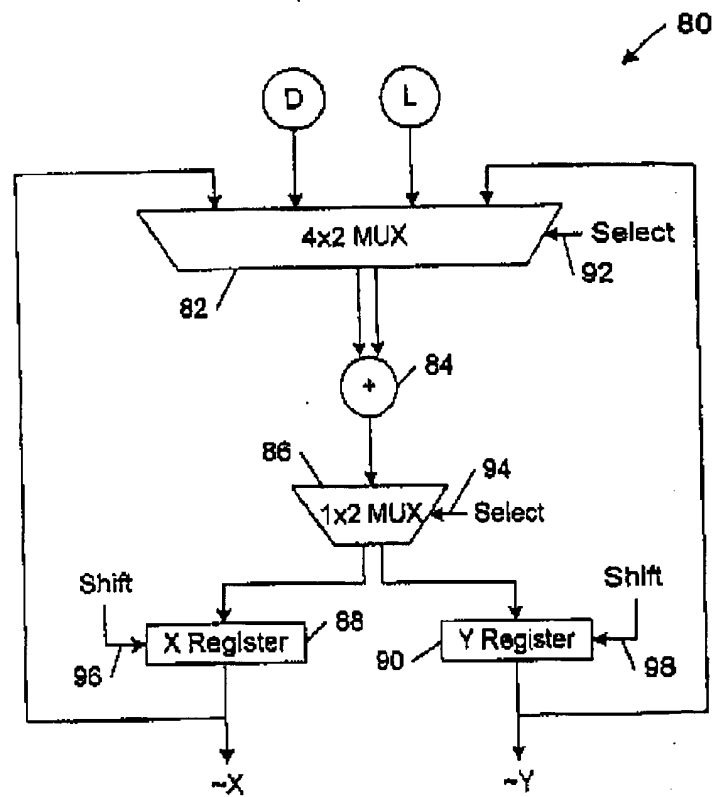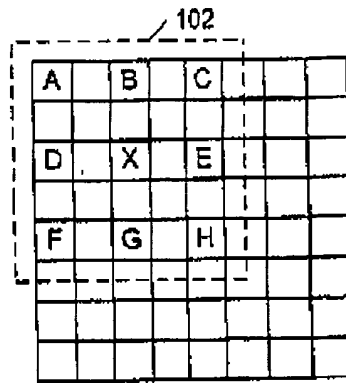FIG. 3A   FIG. 3B   FIG. 3C   FIG. 3D
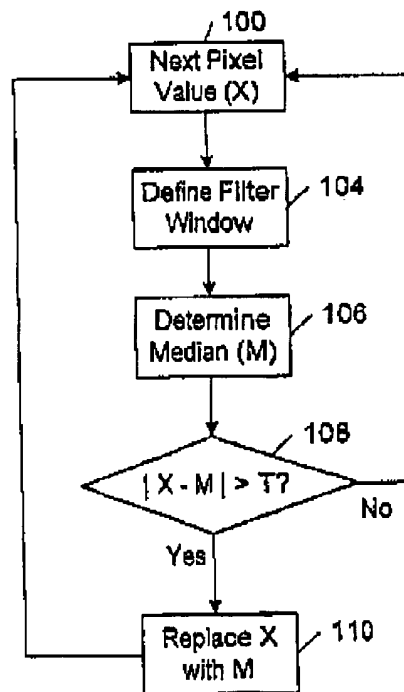
FIG. 4

FIG. 5

FIG. 6

FIG. 7

CORRECTING DEFECT PIXELS IN A DIGITAL IMAGE

TECHNICAL FIELD

This invention relates to systems and methods for correcting defect pixels in a digital image (for example, a digital still image, a digital video image, or both).

BACKGROUND

Digital images are formed from an array of pixels (picture elements), each pixel having an assigned color and intensity. A digital image may be captured by a charged-coupled device (CCD) with a planar array of photodetectors. Each photodetector is located at a respective pixel position. Defective photodetectors produce imperfections in the corresponding pixel locations in the resulting digital image. Although CCD manufacturers have reduced the photodetector defect rate significantly, elimination of all defects typically is not practical. Consequently, manufacturers of digital imaging devices generally include systems designed to correct imperfections caused by photodetector defects. Various correction systems have been proposed. Some correction systems include image processors designed to interpolate the correct values of defect pixels based upon the values of non-defective neighboring pixels. Such correction systems may be classified as directional interpolation systems, or non-directional averaging (or erosion) systems. For example, in one proposed directional interpolation system, a defect pixel value is obtained by linear interpolation of a line of pixels that extends through the defect pixel and is oriented in a direction determined from gradient information derived from the pixels neighboring the defect pixel. Typical non-directional averaging systems tend to be easier to implement; however, they also tend to overly smooth non-edge image areas and significantly distort edges in the corrected image.

SUMMARY

The invention features improved systems and methods of correcting one or more defect pixels in a source image formed from a plurality of pixels each having a respective value.

In one aspect, the invention features a scheme in which a defect pixel is corrected based upon curvature information computed from pixel values located near the defect pixel.

Embodiments may include one or more of the following features.

5   The defect pixel preferably is corrected by interpolating values of pixels aligned with the defect pixel in an interpolation direction determined from the curvature information. The interpolation direction may be determined by comparing curvature values computed for two orthogonal directions. The interpolation direction preferably corresponds to the direction having the lower

10   curvature value. The defect pixel preferably is assigned a value computed from a linear interpolation of values of pixels aligned with the defect pixel along the interpolation direction. The defect pixel may be assigned a value computed from a linear interpolation of values of two non-defective pixels located on opposite sides of the defect pixel. The assigned defect pixel value may be computed from

15   the operations of adding and shifting.

     The defect pixel may be corrected based upon curvature information computed for two different image directions. For example, a horizontal curvature value may be computed based upon pixel values from two groups of horizontally aligned pixels located on opposite sides of the defect pixel. A vertical curvature

20   value may be computed based upon pixel values from two groups of vertically aligned pixels located on opposite sides of the defect pixel. Alternatively, curvature values may be computed based upon values of pixels surrounding the defect pixel. The defect pixel may be corrected based upon a comparison of the computed curvature values. For example, the defect pixel may be corrected by

25   interpolating values of pixels aligned vertically with the defect pixel in response to a determination that the horizontal curvature exceeds the vertical curvature. Alternatively, the defect pixel may be corrected by interpolating values of pixels aligned horizontally with the defect pixel in response to a determination that the vertical curvature exceeds the horizontal curvature.

30   The source image may be obtained from an image sensor having a plurality of light detectors each generating a pixel value of a respective color, wherein the colors are arranged in multiple color planes of non-overlapping pixel locations.

<div align="center">2</div>

The defect pixel may be corrected based upon curvature information computed from values of pixels in the same color plane.

In another aspect, the invention features a scheme in which a median pixel value is determined from values of pixels on the same color plane located near a defect pixel, and the defect pixel is corrected based upon the median pixel value.

Embodiments may include one or more of the following features.

The median pixel value preferably is determined from values of pixels on the same color plane adjacent to the defect pixel. A difference value representative of the difference between the defect pixel value and the computed median pixel value may be computed, and the defect pixel may be corrected by replacing the defect pixel value with the median value in response to a determination that the difference value exceeds a threshold value.

Among the advantages of the invention are the following.

The curvature-based directional interpolation invention provides a robust and efficient scheme for recovering corrupted pixel data in both low frequency (smooth) areas and high frequency (edge and corner) areas of the source image. The median filter invention, on the other hand, provides an efficient scheme for recovering corrupted pixel data on the fly, especially when the locations of defects are not known beforehand. The median filter invention effectively removes outliers (i.e., adjacent defect pixels), while substantially preserving edges and corners in the source image.

Other features and advantages of the invention will become apparent from the following description, including the drawings and the claims.

DESCRIPTION OF DRAWINGS

FIG. 1 is a block diagram of selected components of a digital imaging device.

FIG. 2 is a diagrammatic view of a pixel array with a Bayer geometry color pattern formed from non-overlapping color planes.

FIGS. 3A-3D are diagrammatic views of four pixel array color planes that may be extracted from the Bayer geometry color pattern of FIG. 2.

FIG. 4 is a flow diagram of a method of correcting a defect pixel in a digital source image.

FIG. 5 is block diagram of a circuit for computing a linear interpolation approximation.

FIG. 6 is a diagrammatic view of a median filter window superimposed on a pixel array.

FIG. 7 is a flow diagram of a method of correcting a defect pixel in a digital source image.

DETAILED DESCRIPTION

Referring to FIG. 1, a digital imaging device 10 has an imaging section 12 and a correction filter section 14. Imaging section 12 includes a lens 16, a color filter array 18 (e.g., a Bayer array), a CCD 20, an analog-to-digital (A/D) converter 22, and an image buffer 24 (e.g., a random access memory). Correction filter section 14 includes a digital signal processor (DSP) 26 and a random access memory (RAM) 28. A controller 30 coordinates the operation of the components of imaging section 12 and correction filter section 14. Digital imaging device 10 may be a digital camera configured to record digital still images or a digital video recorder configured to record digital video images. Images captured by digital imaging device 10 may be recorded in a storage memory 32 (e.g., a removable memory card or a digital video cassette). In operation, image light is captured by lens 16 and directed through color filter array 18 to CCD 20. CCD 20 converts impinging filtered light into a plurality of analog image signals, and A/D converter 22 converts the analog image signals into digital image signals. The digital image signals are stored in image buffer 24 as source image data for processing. After a sufficient amount of source image data has been stored, the source image data is processed by DSP 26, which corrects one or more defect pixels in the source image to produce corrected image data. The corrected image data may be processed further (e.g., the source image may be demosaiced), and the final image data may be stored in storage memory 32.

Referring to FIG. 2, color filter array 18 is configured to transmit light of a single color to each pixel location of CCD 20. For example, in a Bayer color filter array geometry, the pixel colors are arranged in multiple color planes of non-overlapping pixel locations. When superimposed, the color planes form a pattern

4

of chrominance colors (red and blue) interspersed among a checkerboard pattern of luminance colors (green), as shown in FIG. 2.

In the present embodiment, DSP 26 corrects defect pixels based upon information computed from the values of the nearest neighbor pixels; other embodiments, however, may incorporate into the correction process the values of pixels located farther away from the defect pixels. A defect pixel may be identified during calibration of digital imaging device 10 using known techniques (e.g., filtering and testing images produced by CCD 20). The locations of defect pixels may be stored as a defect map in RAM 28. In the present embodiment, the nearest neighbors of a particular defect pixel are the eight pixels located in the same color plane as the defect pixel. For example, as shown in FIGS. 3A and 3B, in a red color plane 40 and a blue color plane 42, the eight nearest neighbors of a defect pixel (X, Y) are spaced apart from the defect pixel by one pixel location. Referring to FIGS. 3C and 3D, two different color planes 44, 46 may be extracted from the Bayer geometry to identify the nearest neighbors of a defect pixel (Z). In color plane 44, the eight nearest neighbors are spaced apart from the defect pixel by one pixel location in every direction whereas, in color plane 46, four of the eight nearest neighbors are located adjacent to the defect pixel and the remaining neighbors are spaced apart from the defect pixel by one pixel location.

As mentioned above, in one embodiment, DSP 26 corrects defect pixel values based upon curvature information computed from pixel values located near each of the defect pixels. This approach provides a robust and efficient scheme for recovering corrupted pixel data in both low frequency (smooth) areas and high frequency (edge and corner) areas of the source image.

Referring to FIG. 4, in one embodiment, DSP 26 may correct a defect pixel in a source image as follows. If the curvature information is complete (i.e., none of the nearest neighbors is defective) (step 50), a horizontal curvature value (step 52) and a vertical curvature value (step 54) are computed. The horizontal curvature (HC) and vertical curvature (VC) values may be estimated respectively from two groups of horizontally-aligned and vertically-aligned pixels located on opposite sides of the defect pixel as follows:

$$\text{Horizontal Curvature} = |A - 2 \times B + C| + |F - 2 \times G + H| \tag{1}$$
$$\text{Vertical Curvature} = |A - 2 \times D + F| + |C - 2 \times E + H| \tag{2}$$

where, A, B, C, D, E, F, G and H correspond to the values of the eight nearest
neighbor pixels of a defect pixel X, arranged in accordance with the following
pixel array (i.e., A, B and C are the top neighbors, D is the left neighbor, E is the
right neighbor, and F, G and H are the bottom neighbors).

A		B		C
D		X		E
F		G		H

Pixel Array I: Curvature Information is Complete

If the horizontal curvature exceeds the vertical curvature (step 56), the defect
pixel is corrected by interpolating values of pixels aligned vertically with the
defect pixel (i.e., the value of X is replaced with the value of $\frac{1}{2} \cdot (B + C)$) (step
58). Otherwise, the defect pixel is corrected by interpolating values of pixels
aligned horizontally with the defect pixel (i.e., the value of X is replaced with the
value of $\frac{1}{2} \cdot (D + E)$) (step 60).

If the curvature information is incomplete (step 50) but the gradient
information is complete (step 62), a horizontal gradient value (step 64) and a
vertical gradient value (step 66) are computed. The horizontal gradient (HG) and
vertical gradient (VG) values may be computed as follows:

$$\text{Horizontal Gradient} = |I - J| \tag{3}$$
$$\text{Vertical Gradient} = |I - K| \tag{4}$$

where I, J and K are arranged on different color planes in accordance with the
following pixel array.

A		B		C
	I	J		
D	K	X		E
F		G		Defect

Pixel Array II: Incomplete Curvature Information,
Complete Gradient Information

5   The values of other groups of pixels (e.g., A, B and D, or E, G and H), which are
located at corner positions near the defect pixel, also may be used to compute the
horizontal and vertical gradient information. If the horizontal gradient exceeds
the vertical gradient (step 68) and the vertical interpolation information is
complete (step 70), the defect pixel is corrected by interpolating values of pixels

10   aligned vertically with the defect pixel (i.e., the value of X is replaced with the
value of $\frac{1}{2} \cdot (B + G)$) (step 58). On the other hand, if the vertical gradient
exceeds the horizontal gradient (step 68) and the horizontal interpolation
information is complete (step 72), the defect pixel is corrected by interpolating
values of pixels aligned horizontally with the defect pixel (i.e., the value of X is

15   replaced with the value of $\frac{1}{2} \cdot (D + E)$) (step 60). If the information required for
the horizontal or vertical interpolation is incomplete (steps 70, 72), the defect
pixel value is replaced with a linear interpolation approximation in the
interpolation direction (step 74). For example, if a horizontal interpolation is
called for but both pixel X and pixel Y are defective, the values of pixel X and

20   adjacent pixel Y may be recovered by linearly interpolating between pixel D and
pixel L, as follows.

$$X = 2/3 \times D + 1/3 \times L \tag{5}$$
$$Y = 1/3 \times D + 2/3 \times L \tag{6}$$

25

where D and L are arranged on the same color plane in accordance with the
following pixel array.

7

A		B		C		
D		X		Y		L
F		G		Defect		

Pixel Array III: Incomplete Curvature Information, Complete
Gradient Information, Incomplete Interpolation Information

The system used to compute this linear interpolation approximation may be
simplified by reducing the approximation to a form that may be computed by a
series of adding and shifting operations. For example, equations (5) and (6)
above, may be approximated by the following equations:

$$X \sim \tfrac{3}{4} \times D + \tfrac{1}{4} \times L = \tfrac{1}{2}\cdot(D + \tfrac{1}{2}\cdot(L + D)) \qquad (7)$$

$$Y \sim \tfrac{3}{4} \times L + \tfrac{1}{4} \times D = \tfrac{1}{2}\cdot(L + \tfrac{1}{2}\cdot(D + L)) \qquad (8)$$

Equations (5) and (6) may be approximated with greater accuracy by the
following equations:

$$X \sim 5/8 \times D + 3/8 \times L = \tfrac{1}{2}\cdot(D + \tfrac{1}{2}\cdot(L + \tfrac{1}{4}\cdot(D + L))) \qquad (9)$$

$$Y \sim 5/8 \times L + 3/8 \times D = \tfrac{1}{2}\cdot(L + \tfrac{1}{4}\cdot(D + \tfrac{1}{2}\cdot(L + D))) \qquad (10)$$

In a preferred embodiment, the values of defect pixels X and Y are approximated
by equations (9) and (10), respectively. However, equations (5) and (6) may be
approximated with even greater accuracy by the following equations:

$$X \sim 11/16 \times D + 5/16 \times L = \tfrac{1}{2}\cdot(D + \tfrac{1}{2}\cdot(L + \tfrac{1}{2}\cdot(D + \tfrac{1}{2}\cdot(L + D)))) \qquad (11)$$

$$Y \sim 11/16 \times L + 5/16 \times D = \tfrac{1}{2}\cdot(L + \tfrac{1}{2}\cdot(D + \tfrac{1}{2}\cdot(L + \tfrac{1}{2}\cdot(D + L)))) \qquad (12)$$

The accuracy with which equations (5) and (6) are approximated may be
increased still further by iteratively increasing the number of adding and shifting
operations.

8

Referring to FIG. 5, each of the approximation equations (7)-(12) may be computed by an approximation circuit 80. Approximation circuit 80 includes a 4 × 2 multiplexer (MUX) 82, an adder 84, a 1 × 2 multiplexer 86, and two registers 88, 90. Multiplexer 82 has four inputs: two inputs are coupled to receive the two pixel values (D and L) from which the defect pixel values (X and Y) are to be interpolated; the remaining two inputs are coupled to receive the output values from registers 88 and 90, respectively. A select signal 92 controls which two of the four multiplexer inputs are applied to the inputs of adder 84. The output of adder 84 – which represents the sum of the adder input values – is applied to the input of multiplexer 86. A select signal 94 controls which register 88, 90 receives the sum applied to the input of multiplexer 86.

In operation, controller 30 may choreograph the operation of approximation circuit 80 as follows. First, through select signal 92, controller 30 directs multiplexer 82 to apply inputs D and L to the inputs of adder 84. The resulting sum (L + D) is applied to register 90. Through shift signal 98, controller 30 shifts the contents of register 90 by one decimal place to produce the quantity $\frac{1}{2} \cdot (L + D)$. The quantity in register 90 is applied to the input of multiplexer 82. In the next cycle, controller 30 directs multiplexer 82 to apply inputs D and the quantity of register 90 to the inputs of adder 84. The resulting sum $(D + \frac{1}{2} \cdot (L + D))$ is applied to register 88 and, through shift signal 96, controller 30 shifts the contents of register 88 by one decimal place to produce the sum $(\frac{1}{2} \cdot (D + \frac{1}{2} \cdot (L + D)))$, which corresponds to the first approximation of the interpolation of defect pixel X (i.e., equation (7)). In the following cycle, the quantity of register 90 and input L are applied to the inputs of adder 84. The resulting sum $(L + \frac{1}{2} \cdot (D + L))$ is applied to register 90. Controller 30 shifts the contents of register 90 by one decimal place to produce the quantity $(\frac{1}{2} \cdot (L + \frac{1}{2} \cdot (D + L)))$, which corresponds to the first approximation of the interpolation of defect pixel Y (i.e., equation (8)). The interpolation approximations of equations (9) and (10) may be computed by further cycling of approximation circuit 80. For example, the sum of input D and the contents of register 90 may be applied to register 88, and the contents of register 88 may be shifted by one decimal place to produce the quantity $(\frac{1}{2} \cdot (D + \frac{1}{2} \cdot (L + \frac{1}{2} \cdot (D + L))))$ (i.e., equation (9)). In addition, the sum of input L and the contents of register 88 may be applied to

register 90, and the contents of register 90 may be shifted by one decimal place to produce the quantity $(\frac{1}{2} \cdot (L + \frac{1}{2} \cdot (D + \frac{1}{2} \cdot (L + D))))$ (i.e., equation (10)). Further cycling of circuit 80 in a similar manner would increase the accuracy with which the linear interpolations for X and Y may be approximated.

Referring back to FIG. 4, if both the curvature information (step 50) and the gradient information (step 62) are incomplete, the defect pixel value (X) is replaced with the average value computed from the values of the non-defective nearest neighbors (step 76):

$$X = (P_1 + P_2 + \ldots + P_N)/N \tag{13}$$

where N is the number of non-defective nearest neighbors, and P_i are the values of the non-defective nearest neighbors. In one embodiment, pixels are processed in raster scan order. Consequently, in this embodiment, pixels above and to the left of the current defect pixel (X) are guaranteed to have valid values and equation (13) will produce a valid estimate of the defect pixel value.

In another embodiment, DSP 26 corrects defect pixel values based upon a median pixel value computed from pixel values located near each of the defect pixels. This median filter approach provides an efficient scheme for recovering corrupted pixel data on the fly, especially when the locations of defects are not known beforehand (i.e., a defect map is not stored in memory 28). This approach effectively removes outliers (i.e., adjacent defect pixels), while substantially preserving edges and corners in the source image.

Referring to FIGS. 6 and 7, DSP 26 may be configured to implement a median filter as follows. For each pixel value (X) (step 100), a median filter window 102 is defined (step 104). In this embodiment, defect pixels are corrected based upon information computed from the values of the nearest neighbor pixels; other embodiments, however, may incorporate into the correction process the values of pixels located farther away from the defect pixels. In this embodiment, the nearest neighbors of a particular pixel are the eight pixels (A, B, C, D, E, F, G and H) located in the same color plane as the pixel. The median value (M) of the values of the nearest neighbor pixels is determined (step 106). An eight-input sorting (or ordering) circuit may be used to determine the median value of the

nearest neighbor pixels with the median filter window 102. If the difference between the pixel value (X) and the median value (M) is greater than a threshold value (T) (step 108), the value of pixel X is replaced with the median value (step 110). Otherwise, the value of pixel X is unchanged, and the next pixel is

5 processed (step 100).

The systems and methods described herein are not limited to any particular hardware or software configuration, but rather they may be implemented in any computing or processing environment. Controller 30 preferably is implemented in firmware. Alternatively, however, controller 30 may be implemented in a high

10 level procedural or object oriented programming language, or in assembly or machine language; in any case, the programming language may be a compiled or interpreted language.

Other embodiments are within the scope of the claims.

For example, although horizontal and vertical directions were selected as

15 the primary processing directions (i.e., the directions used to determine the orientation of predominate image structure and to interpolate values for defect pixels), other processing directions (e.g., diagonal or radial image directions), including two or more directions, may be selected.

In another embodiment, horizontal and vertical curvatures may be

20 estimated by incorporating a filter array that includes three-by-three, horizontal curvature and vertical curvature filter arrays with the following coefficient values:

$\frac{1}{4}$		-1		$\frac{1}{4}$
$\frac{1}{2}$		0		$\frac{1}{2}$
$\frac{1}{4}$		-1		$\frac{1}{4}$

Filter Array I: Horizontal Curvature Filter Array

25

¼		½		¼
-1		0		-1
¼		½		¼

Filter Array II: Vertical Curvature Filter Array

By applying filter array I and filter array II to pixel array I (above), the following horizontal and vertical curvature estimates may be obtained:

Horizontal Curvature $= ¼ \cdot (A - 4 \cdot B + C + 2 \cdot D + 2 \cdot E + F - 4 \cdot G + H)$ (14)

Vertical Curvature $= ¼ \cdot (A + 2 \cdot B + C - 4 \cdot D - 4 \cdot E + F + 2 \cdot G + H)$ (15)

Thus, in this embodiment, curvature values are estimated based upon the values of pixels surrounding the defect pixel.

Still other embodiments are within the scope of the claims.

WHAT IS CLAIMED IS:

1.   A method of correcting one or more defect pixels (X) in a source image formed from a plurality of pixels each having a respective value, comprising correcting a defect pixel (X) based upon curvature information computed from pixel values located near the defect pixel (X).

5   2.   The method of claim 1, wherein the defect pixel (X) is corrected based upon curvature information computed for two different image directions.

3.   The method of claim 1, further comprising computing a curvature value based upon values of pixels surrounding the defect pixel (X).

4.   The method of claim 1, wherein the defect pixel (X) is corrected by
10   interpolating values of pixels aligned with the defect pixel (X) in an interpolation direction determined from the curvature information.

5.   The method of claim 4, wherein the defect pixel (X) is assigned a value computed from a linear interpolation of values of pixels aligned with the defect pixel (X) along the interpolation direction.

15   6.   The method of claim 5, wherein the defect pixel (X) is assigned a value computed from a linear interpolation of values of two non-defective pixels located on opposite sides of the defect pixel (X).

7.   The method of claim 6, wherein the assigned defect pixel value is computed from the operations of adding (86) and shifting (88, 90).

20   8.   The method of claim 1, further comprising obtaining the source image from an image sensor (20) having a plurality of light detectors each generating a pixel value of a respective color, wherein the colors are arranged in multiple color planes (40, 42, 44, 46) of non-overlapping pixel locations, and the defect pixel (X) is corrected based upon curvature information computed from
25   values of pixels in the same color plane (40, 42, 44, 46).

9.   A method of correcting one or more defect pixels (X) in a source image formed from a plurality of pixels each having a respective value and corresponding to a respective color plane (40, 42, 44, 46), comprising:

determining a median pixel value (M) from values of pixels on the same color plane (40, 42, 44, 46) located near a defect pixel (X); and

correcting the defect pixel (X) based upon the median pixel value (M).

10.   The method of claim 9, further comprising computing a difference value representative of the difference between the defect pixel value and the computed median pixel value (M), wherein the defect pixel (X) is corrected by replacing the defect pixel value with the median value (M) in response to a determination that the difference value exceeds a threshold value (T).

Application No:	GB 0111466.9	*15*	Examiner:	Rebecca Villis
Claims searched:	1		Date of search:	30 October 2001

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

   UK Cl (Ed.S):

    Int Cl (Ed.7):

Other: Online: EPODOC, WPI, PAJ

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
X	GB 2164521 (SONY) (see fig. 1, page 1 lines 46-73 and page 1 line 121 - page 2 line 78)	1-3
X	GB 2120895 (RCA CORPORATION) (see fig. 3, page 1 line 61-70 and page 1 line 88 - page 2 line 92)	1 and 3
X	EP 0615205 A2 (CANON) (see abstract, page 6 line 45 - page 8 line 45 and page 16 line 38 - page 18 line 8)	1 and 3
X	EP 0478040 A2 (AMPEX) (see fig. 1, col. 3 lines 12-36, col. 4 line 31 - col. 5 line 54 and col. 7 line 58 - col 8 line 43)	1-3